UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/901,368	07/09/2001	Nithyalakshmi Sampathkumar	MS180587.1	6483

27195 7590 10/31/2007

AMIN. TUROCY & CALVIN, LLP
24TH FLOOR, NATIONAL CITY CENTER
1900 EAST NINTH STREET
CLEVELAND, OH 44114

EXAMINER	
HILLERY, NATHAN	
ART UNIT	**PAPER NUMBER**
2176	

NOTIFICATION DATE	DELIVERY MODE
10/31/2007	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

docket1@thepatentattorneys.com
hholmes@thepatentattorneys.com
osteuball@thepatentattorneys.com

Office Action Summary	Application No.	Applicant(s)	
	09/901,368	SAMPATHKUMAR ET AL.	
	Examiner	Art Unit	
	Nathan Hillery	2176	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 August 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-19* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-19* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to communications: RCE filed on 8/13/07.

2. Claims 1 – 19 are pending in the case. Claims 1 and 19 are independent.

Claim Rejections - 35 USC § 101

3. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

4. Claims 1 – 19 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

5. Claims 1 – 19 are directed to transforming data items. This claimed subject matter lacks a practical application of a judicial exception (law of nature, abstract idea, naturally occurring article/phenomenon) since it fails to produce a useful, concrete and tangible result.

Specifically, the claimed subject matter does not produce a tangible result because the claimed subject matter fails to produce a result that is limited to having real world value rather than a result that may be interpreted to be abstract in nature as, for example, a thought, a computation, or manipulated data. More specifically, the claimed subject matter provides for transforming a selective subset of data items. This produced result remains in the abstract and, thus, fails to achieve the required status of having real world value.

Further, the claimed subject matter does not produce a useful result because the claimed subject matter fails to sufficiently reflect at least one practical utility set forth in the descriptive portion of the specification. More specifically, while the described

practical utility is directed to providing for a streaming input and streaming output, incremental XML transformer that incrementally builds output from XML data, the claimed subject matter relates ONLY to transforming the data items.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically teach that or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claim 1, 3, 4, 5 and 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kuznetsov (US 6772413 B2).

8. **Regarding independent claim 1**, Kuznetsov teaches that to transform an input XML vocabulary to another (output) XML vocabulary, the XSLT translator processor must parse the transform, parse the source data, walk the two parse trees to apply the transform, and finally output the data into a stream (Column 14, lines 51 – 59), which meets the limitation of **a transformer that transforms one or more input XML items in a first format to one or more transformed XML items in one or more second XML formats**.

Kuznetsov teaches that any number of translators can be implemented simultaneously, such that an entire set (or selected subset) of packets can be translated during runtime (Column 13, line 66 – column 14, line 1), which meets the limitation of **an output manager that facilitates at least one of selectively pulling and pushing a subset of the one or more input XML items**.

Kuznetsov does not explicitly say that **the subset of the one or more XML items is less than the whole one or more input XML items**.

However Kuznetsov teaches an entire set or selected subset (Column 13, line 66 – Column 14, line 1) as two different entities. Thus, it would have been to a person of ordinary skill in the art to try translating a selected subset that is less than the entire set in an attempt to provide an improved translation, as a person with ordinary skill has good reason to pursue the known options within his or her technical grasp. In turn, because the subset as claimed has the properties predicted by the prior art, it would have been obvious to translate a selected subset that is less than the entire set.

9. **Regarding dependent claim 3**, Kuznetsov teaches that a data translator compiler is adapted for using the XSL stylesheet as its input. The data translator compiler then generates executable machine code that operates as a run-time translator between the source XML and the target XML (Column 14, line 60 – Column 15, line 2), which meets the limitation of **a compiler that compiles one or more style sheets and produce one or more actions that can be employed by the transformer in processing associated with transforming the one or more input XML items**.

10. **Regarding dependent claim 4**, Kuznetsov teaches that an implementation according the present invention may also incorporate predefined functions, or references to external functions that can be called at runtime, according to the needs of the translator, as generated by the translator compiler engine (Column 14, lines 23 –

27), which meets the limitation of **the compiler resolves one or more external references in the one or more style sheets**.

11. **Regarding dependent claim 5**, Kuznetsov teaches that whether the data is recorded in a storage device, preserved in temporary memory, or transmitted over a network, the approach allows many more formats and protocols to be accommodated flexibly while preserving the performance and simplification advantages (Column 9, lines 28 – 33), which meets the limitation of **the input XML items are input from one or more data stores**.

12. **Regarding independent claim 19**, Kuznetsov teaches that to transform an input XML vocabulary to another (output) XML vocabulary, the XSLT translator processor must parse the transform, parse the source data, walk the two parse trees to apply the transform, and finally output the data into a stream (Column 14, lines 51 – 59), which meets the limitation of **a transforming component that transforms an input XML item from a first format to a transformed XML item in one or more second XML formats**.

Kuznetsov teaches that any number of translators can be implemented simultaneously, such that an entire set (or selected subset) of packets can be translated during runtime (Column 13, line 66 – column 14, line 1), which meets the limitation of **an output managing component that facilitates at least one of selectively pulling and**

pushing a subset of the input XML item, the subset of the one or more XML items is less than the whole input XML item.

Kuznetsov teaches that a data translator compiler is adapted for using the XSL stylesheet as its input. The data translator compiler then generates executable machine code that operates as a run-time translator between the source XML and the target XML (Column 14, line 60 – Column 15, line 2), which meets the limitation of **a compiling component that compiles a style sheet and that produces one or more actions that can be employed by the transforming component in processing associated with transforming the input XML item**.

Kuznetsov teaches that whether the data is recorded in a storage device, preserved in temporary memory, or transmitted over a network, the approach allows many more formats and protocols to be accommodated flexibly while preserving the performance and simplification advantages (Column 9, lines 28 – 33) and that the optimization options comprise first optimization pass, which generates intermediate format, and second optimization pass (Column 16, lines 49 – 54), which meets the limitation of **an input abstracting component that presents input XML items stored in one or more different representations to the transforming component in a common representation**.

Kuznetsov teaches that as currently specified by the Worldwide Web Consortium, there are three major components in an XSL processor: XSLT, the transformation engine; Xpath, the node selection and query module; and Formatting Objects, the formatting and end-user presentation layer specification. XML-to-XML data

translation is primarily concerned with the first two modules (Column 14, lines 33 – 39), which meets the limitation of **a node selection abstracting component that dynamically constructs a subset of input XML items from a set of input XML items, the subset of input XML items are responsive to a query**.

13. Claim 2 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kuznetsov (US 6772413 B2) as applied to claim 1 above and further in view of Omoigui (US 20030126136 A1).

14. **Regarding dependent claim 2**, Kuznetsov does not explicitly teach that **the transformer comprises an action frame stack that holds one or more actions, an event state machine that tracks state associated with transforming the one or more input XML items and an event processor that receives events generated in processing the one or more actions stored in the action frame stack**.

However, Omoigui teaches that the system provides support for authentication, authorization, auditing, data privacy, data integrity, availability, and non-repudiation by employing standards such as WS-Security. WS-Security provides a platform for security with XML Web Service applications using standards in the XML Web Service protocol stack. This includes encrypting method calls from clients, support for digital signatures, authenticating the calling user before granting access to an Agency's Semantic Network and XML Web Service methods, etc. (paragraph block 0367), which meets the limitation of **the transformer comprises an action frame stack that holds one or more actions, an event state machine that tracks state associated with transforming the**

one or more input XML items and an event processor that receives events generated in processing the one or more actions stored in the action frame stack.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with the invention of Omoigui because such a combination would provide the readers of Kuznetsov with *an integrated and seamless implementation framework and resulting medium for knowledge retrieval, management, delivery and presentation* (paragraph block 0071).

15. Claims 6 – 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kuznetsov (US 6772413 B2) as applied to claim 1 above and further in view of ADO.NET (English translation).

16. **Regarding dependent claim 6**, Kuznetsov does not explicitly teach that **an input abstracter that exposes data stored in the one or more data stores in a common representation**.

ADO.NET teach that an XpathNavigator is created to abstract data from the xml data set via an XpathNodeIterator by employing a loop (p 19), which meets the limitation of **an input abstracter that exposes data stored in the one or more data stores in a common representation**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

17. **Regarding dependent claim 7**, Kuznetsov does not explicitly teach that **the input abstractor abstracts a reference to a node within an Xpath document**.

ADO.NET teach that an XpathNavigator is created to abstract data from the xml data set via an XpathNodeIterator (p 19), which meets the limitation of **the input abstractor abstracts a reference to a node within an Xpath document**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

18. **Regarding dependent claims 8**, Kuznetsov does not explicitly teach that **the input abstractor exposes the data stored in the one or more data stores as a data model and infoset**.

ADO.NET teach that an XpathNavigator is created to abstract data from the xml data set (p 19), which meets the limitation of **the input abstractor exposes the data stored in the one or more data stores as a data model and infoset**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

19. **Regarding dependent claim 9**, Kuznetsov does not explicitly teach that **the input abstractor provides a cursor model over data stored in a data store to facilitate presenting a stream of nodes to the transformer**.

ADO.NET teach that an XpathNavigator is created to abstract data from the xml data set and sends the data to an XSLT (p 19), which meets the limitation of **the input abstractor provides a cursor model over data stored in a data store to facilitate presenting a stream of nodes to the transformer**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

20. **Regarding dependent claim 10**, Kuznetsov does not explicitly teach that **the input abstractor provides a virtual node that can be employed to traverse the stream of nodes**.

ADO.NET teach that an XpathNavigator is created to abstract data from the xml data set (p 19), which meets the limitation of **the input abstractor provides a virtual node that can be employed to traverse the stream of nodes**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

21. **Regarding dependent claim 11**, Kuznetsov does not explicitly teach that **the input abstractor is an XpathNavigator**.

ADO.NET teach that an XpathNavigator is created to abstract data from the xml data set (p 19), which meets the limitation of **the input abstractor is an XpathNavigator**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

22. **Regarding dependent claim 12**, Kuznetsov does not explicitly teach that **a node selection abstractor that dynamically constructs a subset of input XML items from a set of input XML items, the subset of input XML items are responsive to a query**.

ADO.NET teach that SQL is used to query xml items and store them to an XML data set and that each node in the xml dataset is visited by employing an XpathNodeIterator (pp 18 – 19), which meets the limitation of **a node selection abstractor that dynamically constructs a subset of input XML items from a set of input XML items, the subset of input XML items are responsive to a query**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such

a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

23. **Regarding dependent claim 13**, Kuznetsov does not explicitly teach that **the node selection abstractor facilitates navigating the subset of input XML items**.

ADO.NET teach that each node in the xml dataset is visited by employing an XpathNodeIterator (pp 18 – 19), which meets the limitation of **the node selection abstractor facilitates navigating the subset of input XML items**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

24. **Regarding dependent claim 14**, Kuznetsov does not explicitly teach that **the node selection abstractor is an XpathNodeIterator**.

ADO.NET teach that each node in the xml dataset is visited by employing an XpathNodeIterator (pp 18 – 19), which meets the limitation of **the node selection abstractor is an XpathNodeIterator**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

25. **Regarding dependent claim 15**, Kuznetsov does not explicitly teach that **an optimized data store that stores one or more XML items in a manner that facilitates minimizing processing associated with constructing the subset of input XML items via a query**.

ADO.NET teach that SQL is used to query xml items and store them to an XML data set (pp 18 – 19), which meets the limitation of **an optimized data store that stores one or more XML items in a manner that facilitates minimizing processing associated with constructing the subset of input XML items via a query**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

26. **Regarding dependent claim 16**, Kuznetsov does not explicitly teach that **the optimized data store stores data in a data representation format that facilitates optimizing an Xpath query**.

ADO.NET teach that Xpath document is created and used to store and manipulate the xml data set (pp 18 – 19), which meets the limitation of **the optimized data store stores data in a data representation format that facilitates optimizing an Xpath query**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such

a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

27. **Regarding dependent claim 17**, Kuznetsov does not explicitly teach that **the data representation format comprises expanded XML entities, deleted XML declarations and DOM model data converted to Xpath model data**.

ADO.NET teach that Xpath document is created and used to expand the items in the xml data store so that they can be transformed using an XSLT (pp 18 – 19), which meets the limitation of **the data representation format comprises expanded XML entities, deleted XML declarations and DOM model data converted to Xpath model data**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

28. **Regarding dependent claim 18**, Kuznetsov does not explicitly teach that **the optimized data store is an XpathDocument**.

ADO.NET teach that Xpath document is created and used to store and manipulate the xml data set (pp 18 – 19), which meets the limitation of **the optimized data store is an XpathDocument**.

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Kuznetsov with those of ADO.NET because such a combination would provide the users of Kuznetsov with the benefit of explicit implementation of XPath via source code.

Response to Arguments

29. Applicant's arguments filed 8/18/07 have been fully considered but they are not persuasive.

30. Applicant argues that claims 1 – 19 are statutory under 35 USC 101 because the claims are useful and tangible (p 7 and top of 8).

First, 35 USC 101 requires that a practical application of a judicial exception by way of a physical transformation or a production of a concrete, tangible and useful result. Claim 1 is directed to transforming data items in a computer system. This claimed subject matter lacks a practical application of a judicial exception (law of nature, abstract idea, naturally occurring article/phenomenon) since it fails to produce a useful, concrete and tangible result. Claim 1 constitutes a judicial exception because the invention is a computer-implemented method, which is a set of mathematical algorithms or simply an abstract idea.

Physical transformation occurs when the claimed invention transforms an article or physical object to a different structural state or thing. Physical transformation is an indication that the claim is statutory because such a transformation itself is a useful, tangible and concrete result. However, data transformation is not a physical transformation. Data, by definition, is intangible; so the claim must go further to have a

tangible result. Thus, manipulation of data in a computer is not, in and of itself, sufficient for establishing that a claim is statutory. Likewise, a physical act is not necessarily a physical transformation.

In response to applicant's argument as to why the claimed invention is not tangible (p 7, second paragraph), the claimed subject matter does not produce a tangible result because the claimed subject matter fails to produce a result that is limited to having real world value rather than a result that may be interpreted to be abstract in nature as, for example, a thought, a computation, or manipulated data. More specifically, the claimed subject matter provides for transforming data items. This produced result remains in the abstract and, thus, fails to achieve the required status of having real world value.

In response to applicant's argument as to why the claim is not useful (p 8, first full paragraph), the claimed subject matter does not produce a useful result because the claimed subject matter fails to sufficiently reflect at least one practical utility set forth in the descriptive portion of the specification. More specifically, while the described practical utility is directed to providing for a streaming input and streaming output, the claimed subject matter relates ONLY to transforming the data items.

31. Applicant argues that Omoigui fails to teach that the transformer comprises an action frame stack that holds one or more actions, an event state machine that tracks state associated with transforming the one or more input XML items and an event processor that receives events generated in processing the one or more actions stored in the action frame stack (pp 10 and 11).

The Office disagrees.

Omoigui teaches that the system provides support for authentication, authorization, auditing, data privacy, data integrity, availability, and non-repudiation by employing standards such as WS-Security. WS-Security provides a platform for security with XML Web Service applications using standards in the XML Web Service protocol stack. This includes encrypting method calls from clients, support for digital signatures, authenticating the calling user before granting access to an Agency's Semantic Network and XML Web Service methods, etc. (paragraph block 0367).

Specifically, Omoigui teaches a specific type of action frame stack or at least a stack that is synonymous to and/or performs the same functionality as the claimed action frame stack. Applicant's arguments appear to focus on the lack of specific words used by the reference instead of explaining what differs and how the claims overcome the disclosure of the reference. The disclosure of Omoigui included the cited portion teach encrypting method calls, authenticating a user before granting access, etc, which meet tracking states, processing events and/or one or more actions. It should be noted that the claims only require the ability to hold one action in a 'stack'.

32. Applicant generally argues claims 6 – 10 and 12 – 17 as not being taught because the reference lacks certain key words (pp 11 – 13).

33. Conveniently, applicant fails to mention that claims 11 and 18 provide evidence that claims 6 – 10 and 12 – 17 are met by the reference because ADO.NET teaches a XpathNavigator and XpathDocument, respectively, which is all the claims require.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Nathan Hillery whose telephone number is (571) 272-4091. The examiner can normally be reached on M - F, 10:30 a.m. - 7:00 p.m.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Doug Hutton can be reached on (571) 272-4137. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

NH